INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Strategies for getting the highest likelihood in mixture models

Christophe Biernacki — Gilles Celeux — Gérard Govaert

N° 4255

Septembre 2001

THÈME 4

ISSN 0249-6399 ISRN INRIA/RR--4255--FR

INRIA
RHÔNE-ALPES

Strategies for getting the highest likelihood in mixture models

Christophe Biernacki, Gilles Celeux , Gérard Govaert

Thème 4 — Simulation et optimisation
de systèmes complexes
Projet IS2

Rapport de recherche n° 4255 — Septembre 2001 — 20 pages

Abstract: We compare simple strategies to get maximum likelihood parameter estimation in mixture models when using the EM algorithm. All considered strategies are aiming to initiate the EM algorithm in a good way. They are based on random initialisation, using a Classification EM algorithm (CEM), a Stochastic EM algorithm (SEM) or previous short runs of EM itself. They are compared in the context of multivariate Gaussian mixtures on the basis of numerical experiments on both simulated and real data sets. The main conclusions of those numerical experiments are the following. The simple random initialisation which is probably the most employed way of initiating EM is often outperformed by strategies using CEM, SEM or shorts runs of EM before running EM. Thus, those strategies can be preferred to the random initialisation strategy. Also, it appears that repeating runs of EM is generally profitable since using a single run of EM can often lead to suboptimal solutions. Otherwise, none of the experimented strategies can be regarded as the best one and it is difficult to characterize situations where a particular strategy can be expected to outperform the other ones. However, the strategy initiating EM with repeated short runs of EM can be recommended. This strategy, which as far as we know was not used before the present study have some advantages. It is simple, performs well in a lot of situations presupposing no particular form of the mixture to be fitted to the data and seems little sensitive to noisy data.

Key-words: Multivariate Gaussian Mixture, Optimisation, Initialisation Strategies, EM algorithm, Classification EM, Stochastic EM.

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe, 38330 Montbonnot-St-Martin (France)
Téléphone : +33 4 76 61 52 00 —Télécopie +33 4 76 61 52 52

Stratégies d'obtention de la plus grande vraisemblance dans des modèles de mélanges

Résumé : Nous comparons des stratégies simples pour obtenir l'estimateur du maximum de vraisemblance d'un mélange par l'algorithme EM. Ces stratégies visent à bien initialiser l'algorithme EM. Elles sont fondées sur une initilisation au hasard, l'usage de versions classifiantes ou stochastiques de l'algorithme EM, ou encore sur l'utilisation préalable de courtes exécutions de l'algorithme EM lui-même. Nous les comparons dans le contexte des mélanges gaussiens multivariés par des expérimentations numériques sur des données simulées et réelles. Les principales conclusions sont les suivantes. L'initialisation au hasard qui est certainement la stratégie la plus répandue est souvent battue par les autres qui peuvent lui être préférées. De plus, il s'avère que la répétition d'exécutions des procédures est généralement bénéfique car une unique exécution peut souvent aboutir à une solution sous-optimale. Sinon, aucune des stratégies envisagées ici peut être considérée meilleure que les autres et il est difficile de cerner des situations où une stratégie particulière est censée se comporter mieux que les autres. Cependant, nous recommendons la stratégie qui consiste à initialiser l'algorithme EM par de courtes exécutions préalables de lui-même. Cette stratégie, qui est nouvelle, a plusieurs avantages. Elle est simple, marche souvent bien dans des situations très variées et semble peu sensible à des données bruitées.

Mots-clés : mélange gaussien multidimensionnel, optimisation, stratégies d'initialisation, algorithmes EM, EM classifiant, EM stochastique.

1 Introduction

In most applications, parameters of a mixture model are estimated by maximizing the likelihood and the standard tool to maximum likelihood (ML) estimation for mixture models is the EM algorithm. But EM solution can highly depend of its starting position especially in a multivariate context. This jeopardizes statistical analysis of mixture for two reasons. First ML estimation is expected to provide sensible estimates of the mixture parameters. Secondly, the highest maximized likelihood enters the definition of numerous criteria aiming to select a good mixture model and especially to choose a relevant number of mixture components. Thus, it is important to get the highest criterion value when estimating the parameters of a mixture through maximum likelihood. Let us illustrate this fact with a simple example. We consider a sample of size $n = 50$ from a two-component univariate Gaussian mixture with proportions $p_1 = p_2 = 0.5$, means $\mu_1 = -0.8$, $\mu_2 = 0.8$ and variances $\sigma_1^2 = 1$, $\sigma_2^2 = 1.5$. All the parameters are supposed to be known, except the means μ_1 and μ_2. The likelihood has two local maxima as shown in Figure 1.

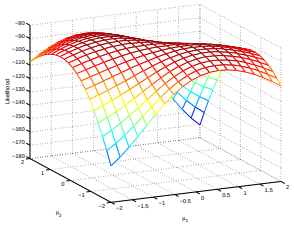


Figure 1: A two-mode likelihood surface

If the lowest likelihood maximum is selected, it can have consequence for choosing the number of components K. For instance, Table 1 gives the AIC criterion values (Akaike 1974) for $K = 1$ and for the two different ML solutions for $K = 2$. Thus, despite its marked tendency to favor too complex models, AIC concludes wrongly for a single Gaussian distribution when the lowest local maximum likelihood is selected.

In this paper, we present and compare simple strategies, on the basis of numerical experiments, to deal with the problem of getting the highest likelihood value in the framework of multivariate Gaussian mixtures. We pay attention to this particular mixture model for simplicity and because it is by far the most employed mixture model with many applications in cluster analysis and statistical pattern recognition (see Mclachlan and Peel 2000).

	1 comp.	2 comp. highest ML	2 comp. lowest ML
AIC	-85.29	-84.88	-85.95

Table 1: AIC criterion values for different ML estimates

Moreover, EM can be thought of as to be much sensitive to initial position in a multivariate context. The strategies we consider are twofold:

- They are acting on the starting values by using a Classification EM (CEM) algorithm, a Stochastic EM (SEM) algorithm or short runs of the standard EM algorithm.
- They are repeating algorithms as EM or some combination of CEM and EM from different initial positions.

The CEM algorithm tends to produce a mixture with well separated components which can be regarded as a good initial solution for EM. The SEM algorithm by using random drawing at each iteration prevents from converging to the first stationary point of the loglikelihood it encounters. For this reason it can avoid sub-optimal maxima of the loglikelihood function.

The paper is organized as follows. The setting of our study and the material we need to define the competing strategies are presented in Section 2. In particular, we present the Classification and the Stochastic EM for mixtures and give a description of the main features of the software MIXMOD devoted to the statistical analysis of Gaussian mixtures. This software MIXMOD coded in in C++ with an interface with Scilab (`http://www.inrialpes.fr/is2/pub/software/MIXMOD`) has made this extentive experimental study possible. Our strategies are presented and slighty discussed in Section 3. Numerical experiments are presented in Section 4: In Section 4.1 the game rules of the experiments are detailed and extensive numerical experiments on both simulated and real data are presented in Section 4.2. The conclusions of our study are drawn in Section 5.

2 Gaussian mixture maximum likelihood estimation

Gaussian mixture model is a powerful model for clustering, pattern recognition and multivariate density estimation (see for instance the monograph of McLachlan and Peel 2000). In this model, data $\mathbf{x}_1, \ldots, \mathbf{x}_n$ in $\mathbf{R}^d$ are assumed to arise from a random vector with density

$$f(\mathbf{x}) = \sum_{k=1}^{K} p_k \Phi(\mathbf{x}|\mu_k, \Sigma_k) \tag{1}$$

where the p_k's are the mixing proportions ($0 < p_k < 1$ for all $k = 1, \ldots, K$ and $\sum_k p_k = 1$) and $\Phi(\mathbf{x}|\mu, \Sigma)$ denotes the density of a Gaussian distribution with mean vector μ and variance matrix Σ. Generally, the mixture parameters

$$\theta = (p_1, \ldots, p_K, \mu_1, \ldots, \mu_K, \Sigma_1, \ldots, \Sigma_K)$$

are estimated by maximizing the loglikelihood

$$L(\theta|\mathbf{x}_1, \ldots, \mathbf{x}_n) = \sum_{i=1}^{n} \ln \left[\sum_{k=1}^{K} p_k \Phi(\mathbf{x}_i|\mu_k, \Sigma_k) \right]. \tag{2}$$

The EM algorithm. The standard tool for finding maximum likelihood solution is the EM algorithm (Dempster, Laird and Rubin 1977). In our context, it can be sketched as follows. Starting from an initial parameter θ^0, an iteration of the EM algorithm consists of two steps.

- **E step:** The current conditional probabilities $t_k(\mathbf{x}_i)(1 \leq i \leq n, 1 \leq k \leq K)$ that $\mathbf{x}_i$ arises from the kth mixture component for the current value of θ are computed according to the equation

$$t_k(\mathbf{x}_i) = \frac{\hat{p}_k \Phi(\mathbf{x}_i | \hat{\mu}_k, \hat{\Sigma}_k)}{\sum_{\ell=1}^{K} \hat{p}_\ell \Phi(\mathbf{x}_i | \hat{\mu}_\ell, \hat{\Sigma}_\ell)}. \tag{3}$$

- **M step:** The ML estimates $\hat{p}_k, \hat{\mu}_k, \hat{\Sigma}_k$ are updated using the conditional probabilities $t_k(\mathbf{x}_i)$ as conditional mixing weights. Detailed formulas are not given here. They can be found for the models we consider in this paper in Celeux and Govaert (1995).

The characteristics of the EM algorithm are well documented (see for instance McLachlan and Krishnam 1997). It leads in general to simple equations, has the nice property of increasing the loglikelihood at each iteration until stationarity, and in many circunstances, it derives sensible parameter estimates and consequently it is a popular tool to derive maximum likelihood estimation. However, EM is known to converge slowly in some situations. We do not address this important aspect of EM here. It has received much attention recently and many algorithms aiming to speed up the convergence of EM while preserving its simplicity have been proposed (see the chapter 4 of McLachlan and Krishnam 1997, and, for algorithms specific to the mixture context Liu and Sun 1997 and Celeux, Chrétien, Forbes and Mkhadri 2001). The second important drawback of EM is that its solution can highly depend of its starting position and consequently produce sub-optimal maximum likelihood estimates. The strategies that we propose are aiming to overcome this limitation. Note that both drawbacks, slow convergence and dependence from initial positions, can be regarded as linked in practical situations. Actually, it is possible that starting from some position leads to a slower convegence rate for EM and that EM is stopped before reaching a sensible value of the likelihood. To act against this high dependency of EM on its initial position, we make use of related algorithms, CEM and SEM, that we present now.

The CEM algorithm. This algorithm (see Celeux and Govaert 1992) incorporates a classification step between the E and and M steps of EM. Starting from an initial parameter θ^0, an iteration of CEM consists of three steps.

- **E step:** The conditional probabilities $t_k(\mathbf{x}_i)(1 \leq i \leq n, 1 \leq k \leq K)$ for the current value of θ are computed according to (3).
- **C step:** A partition $P = (P_1, \ldots, P_K)$ of $\mathbf{x}_1, \ldots, \mathbf{x}_n$ is designed by assigning each point $\mathbf{x}_i$ to the component maximizing the conditional probability $t_k(\mathbf{x}_i)$ according to the Maximum a Posteriori (MAP) principle.

- **M step:** The ML estimates $(\hat{p}_k, \hat{\mu}_k, \hat{\Sigma}_k)$ are computed using the cluster P_k as subsample $(1 \leq k \leq K)$ of the kth mixture component.

The main features of CEM are the followings. CEM is a *K-means*-like algorithm and contrary to EM, it converges in a finite number of iterations. CEM is not maximizing the observed likelihood (2) but aims maximizing in θ and $\mathbf{z}_1, \ldots, \mathbf{z}_n$ the complete loglikelihood CL where the missing component label $\mathbf{z}_i$ of each sample point is included in the data set:

$$CL(\theta|\mathbf{z}_1, \ldots, \mathbf{z}_n, \mathbf{x}_1, \ldots, \mathbf{x}_n) = \sum_{k=1}^{K} \sum_{i/z_i=k} \ln [p_k \Phi(\mathbf{x}_i|\mu_k, \Sigma_k)]. \quad (4)$$

As a consequence, CEM provides biased estimates of the parameters especially when the mixture components are overlapping (see for instance Celeux and Govaert 1993).

The SEM algorithm. This algorithm (see for instance Celeux, Chauveau and Diebolt 1996) is a stochastic algorithm incorporating between the E and M steps a restoration of the unknown component labels $\mathbf{z}_i$, $i = 1, \ldots, n$, by drawing them at random from their current conditional distribution. Starting from an initial parameter θ^0, an iteration of SEM consists of three steps.

- **E step:** The conditional probabilities $t_k(\mathbf{x}_i)(1 \leq i \leq n, 1 \leq k \leq K)$ for the current value of θ are computed according to (3).
- **S step:** A partition $P = (P_1, \ldots, P_K)$ of $\mathbf{x}_1, \ldots, \mathbf{x}_n$ is designed by assigning each point $\mathbf{x}_i$ at random to one of the mixture components according to the multinomial distribution with parameter $(t_k(\mathbf{x}_i), 1 \leq k \leq K)$.
- **M step:** The ML estimates $(\hat{p}_k, \hat{\mu}_k, \hat{\Sigma}_k)$ are computed using the cluster P_k as subsample $(1 \leq k \leq K)$ of the kth mixture component.

The main features of SEM are the followings. SEM is a Data Augmentation algorithm (see Wei and Tanner 1990). SEM does not converge pointwise. It generates a Markov chain whose stationary distribution is more or less concentrated around the ML parameter estimator. A natural parameter estimate from a SEM sequence $\{\theta^r\}$ is the mean of the iterates values obtain after a *burn-in* period. An alternative estimate is to consider the parameter value leading to the highest likelihood in a SEM sequence.

We now present different Gaussian mixture models of practical interest that we use in this experimental study. Those models are based on the eigenvalue decomposition of the component variance matrices.

The variance matrix eigenvalue decomposition Banfield and Raftery (1993) have considered a parametrization of the variance matrix Σ_k of the kth component in terms of its eigenvalue decomposition, $\Sigma_k = \lambda_k D_k A_k D_k'$, where λ_k defines the volume of P_k, D_k is an orthogonal matrix which defines its orientation and A_k is a diagonal matrix with determinant

1 which defines its shape. This eigenvalue decomposition can model various situations. First, we can allow the volumes, the shapes and the orientations of variance matrices to vary or to be equal between components. Variations on assumptions on the parameters λ_k, D_k and A_k $(1 \leq k \leq K)$ lead to eight general models of interest. Moreover two other families of situations are of interest. Secondly, we can assume that the Σ_k's are diagonal matrices. This particular parametrization gives rise to four additionnal models. The last family of models consists of assuming spherical shapes, namely $A_k = I$, I denoting the identity matrix. It leads to two additionnal models according to the assumption regarding the volume of the mixture components. Finally, we get 14 different models for which the ML equations appearing in the M step of EM and of CEM and SEM can be found in Celeux and Govaert (1995).

The MIXMOD software We coded in C++ with an interface with Scilab, a software devoted to the identification of Gaussian multivariate mixtures. This software has the following features:

- Parameter estimation can be performed using EM, CEM and SEM or any combination of these algorithms with any number of iterations for each algorithm.
- It allows the estimation of the fourteen models derived from the Banfield-Raftery's eigenvalue decomposition of variance matrices.
- Assessing models is possible using several criteria including BIC (approximation of the integrated likelihood), ICL (approximation of the completed integrated likelihood), NEC (entropy criterion). (See McLachlan and Peel 2000 chapter 6, for a presentation of those criteria among others.)

In the present study, we are not concerned with the last aspect. We are essentially aiming to take profit of the various possibilities to combine the algorithms EM, CEM and SEM in the MIXMOD software to derive simple strategies to get a sensible maximum likelihood value.

3 Experimented strategies

In this section we present the strategies that we consider in our numerical experiments. Some of those strategies are standard strategies and consist essentially of initializing the EM algorithm from random positions. Two of them initialize EM from solutions derived from the CEM algorithm and two other strategies initialize EM from SEM solutions. And finally, two strategies are starting from short runs of EM before running the whole EM algorithm. The principles leading to the definitions of those strategies are the following: we just want take profit of the facilities available in MIXMOD to combine the EM, CEM and SEM algorithms in any order to get sensible and simple ways of initiating EM in a multivariate general context. We do not consider sophisticated data analysis tools to deal with the initiating problem of EM (see for instance Ueda and Nakano 1998). In our opinion,

such strategies can give good results on specific examples, but are painfully slow and may be not beneficial in a general context (see McLachlan and Peel 2000 chapter 2, section 4).

Random inialization The standard way to initiate the EM algorithm consists of initializing it from a random position. Usually, especially in a multivariate context, this random initial position is obtained by drawing at random centres in the data set. Since this is probably the most employed way of initiating EM, it can be regarded as a reference strategy. It is denoted "1EM" in the following. An extension of this simple strategy consists of repeating it x times from different random positions and selecting the solution maximizing the likelihood among those x runs. We denote "xEM" this strategy.

Using the CEM algorithm In many circumstances and especially in a cluster analysis setting, the mixture component means are expected to be different. Thus, a reasonable and largely employed way of initiating EM consists of starting from the solution of a K-means type algorithm. Acting in such a way, it is hoped that the initial position will be sensible. This point of view leads to the following strategies. Runs of CEM from random positions followed by EM from the position providing the highest *completed* loglikelihood obtained with CEM. For each CEM run, we consider the same assumptions on the mixture components that we consider when running EM. We denote "1CEM-EM" this strategy. And, x repetitions of the previous strategy give rise to an additional strategy denoted "xCEM-EM".

Using short runs of EM One advantage of initiating EM with CEM lies in the fact that CEM converges generally in a small number of iterations. Thus, without consuming a large amount of CPU times, several runs of CEM can be performed before passing to EM with the best solution among those CEM runs. But, it is possible to recover this feature with short runs of EM. By a short run of EM, we mean that we do not wait for convergence and that we stop the algorithm as soon as

$$\frac{L^q - L^{q-1}}{L^q - L^0} \leq 10^{-2}, \tag{5}$$

L^q denoting the observed loglikelihood at qth iteration. Here 10^{-2} represents a threshold value which has to be chosen on a pragmatic ground. From our experiments, we chose this value to get a number of iterations for such an EM short run approximatively equal to the number of iterations obtained with CEM. It leads to the following strategies. Several short runs of EM from random positions followed by a long run of EM from the solution maximizing the *observed* loglikelihood. We denote "1em-EM" this strategy. And, x repetitions of the previous strategy lead to the so called "$x$em-EM" strategy.

Using Stochastic EM The stochastic EM algorithm generates an ergodic Markov chain. Thus a sequence of parameter estimates via SEM is expected to visit the whole parameter

space with long sojourns in the neighborhood of sensible maxima of likelihood functions. This characteristic of SEM invites to use the following strategies.

- A run of SEM, followed by a run of EM from the solution obtained by computing the mean values of the sequence of parameter estimates provided by SEM after a burn-in period. We denote "SEMmean-EM" this strategy. The idea underlying this strategy is that SEM is expected to spend most of the time near sensible likelihood maxima with a large attractive neighborhood.
- The *same* run of SEM followed by a run of EM from the position leading to the highest maximum likelihood value reached by the SEM sequence of parameter iterates. Here, the idea is that an SEM sequence is expected to enter rapidly in the neighboorhood of the global maximum of the likelihood function. We denote "SEMmax-EM" this strategy.

4 Numerical experiments

As pointed out in Meila and Heckerman (2001), we should not expect to find an initialization strategy that outperforms all the others on all data sets. We simply hope to find honest initialization strategies working well for large classes of situations arising in practice. The only way to answer this question is to perform extensive numerical experiments for various data sets. We realized such experiments on both simulated and real data sets that we summarize in Section 4.2. Before that we precise in Section 4.1 the way those numerical experiments have been managed.

First, it must be noticed that getting parameter estimates maximizing the likelihood for a general Gaussian mixture model is an ill-posed problem since the likelihood is unbounded and the EM algorithm can lead to spurious solutions. (Typically, one of the estimated mixture component has a variance matrix near the null matrix.) Avoiding such spurious solutions must be done on a subjective ground and can be controversial (see McLachlan and Peel 2000 chapter 3 section 10). For this very reason, it can not be included in a Monte Carlo simulation study. In our numerical experiments, we did not deal with this difficult problem. We restrict attention to variance matrices with free shapes and orientations but equal volumes. Acting in such a way, we consider the most general model avoiding unbounded likelihood (see the Appendix for a proof).

4.1 The game rules

Available CPU time. Recall that we are essentially interested in finding strategies leading to the maximum likelihood estimate most often. We are not interested to find strategies leading the faster to a local maximum likelihood. Thus to compare the strategies in competition we proceed as follows. We suppose the user accepts to wait a time t to get the solution. Since CPU time needed to perform an iteration of EM, CEM or SEM algorithms is almost constant, we assume for simplicity that t, the price to be paid for any strategy,

can be converted in the total number of iterations. It means that each strategy will have the same total number of available iterations. Moreover, iterations repartition is equal for each run and each algorithm inside a run. For instance, suppose that 1000 iterations are available. For the eight strategies they are shared in the following way, taking $x = 10$ for the strategies including repeated algorithm runs.

- 1EM: 1000 iterations for EM.
- 10EM: 100 iterations for each EM run.
- 1CEM-EM: 500 iterations for CEM and 500 iterations for EM.
- 10CEM-EM: 10 repetitions of 50 iterations for CEM and 50 iterations for EM.
- 1em-EM: 500 iterations for em and 500 iterations for EM.
- 10em-EM: 10 repetitions of 50 iterations for em and 50 iterations for EM run.
- SEMmean-EM and SEMmax-EM: 500 iterations for SEM and 500 iterations for EM.

Stopping rules. The EM algorithm is stopped with the number of iterations. We do not use threshold to stop a full EM run because possible slow convergence of EM makes this stopping rule hazardous. But short runs of EM in strategies em-EM are stopped using the criterion (5) and a new short run of EM is started again at random until no more iteration is available for this step. The CEM algorithm is stopped when the completed loglikelihood had reached stationarity, and started again until no more iteration is available for this algorithm. The solution providing the largest completed loglikelihood is then selected to initiate EM. Obviously, the stopping rule for the SEM algorithm is the total number of iterations.

Initial conditions. Initial proportions are equal, random initial centres are drawn in the data set, and initial variance matrices are diagonal with diagonal terms containing the empirical variance of the variables.

4.2 Results summary

Monte Carlo experiments. Initial proportions are equal, random initial centres are drawn in the data set, and initial variance matrices are diagonal with diagonal terms containing the empirical variance of the variables.

4.3 Results summary

Monte Carlo experiments. Six types of data in $\mathbf{R}^2$ have been considered. Data P1 arose from a two well-separated component Gaussian mixture with $p_1 = p_2 = 0.5$, $\mu_1 = (0,0)'$, $\mu_2 = (2.5,0)'$, and diagonal variance matrices with $\mathrm{diag}(\Sigma_1) = (3,1/3)$ and $\mathrm{diag}(\Sigma_2) = (1/3,3)$. Data P2 arose from a two poorly separated component Gaussian mixture with

$p_1 = 0.7$, $p_2 = 0.3$, $\mu_1 = \mu_2 = (0,0)'$, and diagonal variance matrices with $\text{diag}(\Sigma_1) = (3, 1/3)$ and $\text{diag}(\Sigma_2) = (1/3, 3)$. Data P3 arose from a four component Gaussian mixture with $p_1 = p_2 = p_3 = p_4 = 0.25$, $\mu_1 = (0,-2)'$, $\mu_2 = (2,0)'$, $\mu_3 = (0,2)'$, $\mu_4 = (-2,0)'$, and diagonal variance matrices with $\text{diag}(\Sigma_1) = \text{diag}(\Sigma_3) = (3, 1/3)$ and $\text{diag}(\Sigma_2) = \text{diag}(\Sigma_4) = (1/3, 3)$. Three alternative data P1noise, P2noise and P3noise are designed by adding noisy data arising from a uniform distribution $[-0.8, +0.8] \times [-0.8, +0.8]$ with proportion 0.2. For each type of data, we generated 30 samples of size $n = 200$. In our experiments, the algorithms were run with two components for P1, P2, P1noise and P2noise and with four components for P3 and P3noise. Moreover, we used the strategies with various numbers of iterations ITEMAX (60, 120 240, 480, 960...until 15360). In what follows, we distinguish *small* number of iterations ITEMAX < 960) and *large* number of iterations (ITEMAX $\geq$ 960).

Table 2 provides the percentage of times a *small* or a *large* number of iterations leads to a higher likelihood for each strategy. Table 3 provides the percentage of times a single run or repeating runs leads to a larger likelihood for strategies EM, CEM-EM and em-CEM, and the percentage of times SEMmean-EM or SEMmax-EM leads to a larger likelihood for strategies SEM-EM for all data sets. It shows that using a large number of iterations is highly preferable. Then Tables 4-9 provide for each data structure and for *large* number of iterations comparisons by pairs of the best representant of each kind of strategy, namely 10Em, 10CEM-EM, 10em-EM and SEMmax-EM. In those tables each row gives the score of a method against all the others, the mean (mean ML) and standard deviation (std ML) of its maximum loglikelihood. For instance, from Table 5 concerning P1noise it appears that 10EM performs better than 10CEM-EM 38 times and perfoms worse 27 times.

For data sets without noise, 10CEM-EM is outperformed by the three other strategies which provide similar results. But, the difference between 10EM-CEM and the other strategies decreases with the component separation. For noisy data, conclusions are not so clear: 10em-EM can be preferred to the other strategies except for data P3noise where 10CEM-EM outperforms all the other strategies.

	EM		CEM-EM		em-EM		SEM-EM	
nb. it.	1	10	1	10	1	10	mean	max
small	7	0	6	4	0	0	6	4
large	35	98	33	87	45	98	46	41

Table 2: Percentage of times a small or a large number of iterations leads to a higher likelihood for each strategy.

Real data sets. The first example on real data sets concerns a population of 2370 stars described by their velocity U toward the galactic center and their velocity V toward the galactic rotation (see Celeux and Govaert 1995). The second example concerns data on 272 eruptions of the Old Faithful geyser in Yellowstone National Park. Each observation

	EM		CEM-EM		em-EM		SEM-EM	
nb. it.	1	10	1	10	1	10	mean	max
small	79	21	67	29	88	12	11	46
large	5	36	5	20	10	12	0	6

Table 3: Percentage of times a single run or repeated runs strategies and SEMmean-EM or SEMmax-EM strategies leads to a higher likelihood.

	10EM	10CEM-EM	10em-EM	SEMmax-EM	mean ML	std ML
10EM	0-0	2-0	0-0	0-0	-659.81	(14.58)
10CEM-EM	0-2	0-0	0-2	0-2	-659.81	(14.58)
10em-EM	0-0	2-0	0-0	0-0	-659.81	(14.58)
SEMmax-EM	0-0	2-0	0-0	0-0	-659.81	(14.58)

Table 4: Comparison of strategies by pairs with a large number of iterations for data set P1.

	10EM	10CEM-EM	10em-EM	SEMmax-EM	mean ML	std ML
10EM	0-0	38-27	8-28	43-8	-909.21	(13.14)
10CEM-EM	27-38	0-0	6-40	54-30	-909.93	(12.54)
10em-EM	28-8	40-6	0-0	57-10	-908.33	(12.33)
SEMmax-EM	8-43	30-54	10-57	0-0	-911.08	(13.88)

Table 5: Comparison of strategies by pairs with a large number of iterations for data set P1noise.

	10EM	10CEM-EM	10em-EM	SEMmax-EM	mean ML	std ML
10EM	0-0	19-0	1-0	0-0	-616.07	(17.81)
10CEM-EM	0-19	0-0	0-19	0-19	-617.94	(18.89)
10em-EM	0-1	19-0	0-0	0-1	-616.07	(17.81)
SEMmax-EM	0-0	19-0	1-0	0-0	-616.07	(17.81)

Table 6: Comparison of strategies by pairs with a large number of iterations for data set P2.

	10EM	10CEM-EM	10em-EM	SEMmax-EM	mean ML	std ML
10EM	0-0	30-42	7-43	43-21	-881.71	(17.34)
10CEM-EM	42-30	0-0	2-30	61-21	-881.23	(18.38)
10em-EM	43-7	30-2	0-0	63-6	-880.18	(17.59)
SEMmax-EM	21-43	21-61	6-63	0-0	-883.67	(17.76)

Table 7: Comparison of strategies by pairs with a large number of iterations for data set P2noise.

	10EM	10CEM-EM	10em-EM	SEMmax-EM	mean ML	std ML
10EM	0-0	36-0	5-1	5-3	-754.28	(13.23)
10CEM-EM	0-36	0-0	1-35	0-35	-755.64	(13.32)
10em-EM	1-5	35-1	0-0	5-7	-754.28	(13.23)
SEMmax-EM	3-5	35-0	7-5	0-0	-754.29	(13.24)

Table 8: Comparison of strategies by pairs with a large number of iterations for data set P3.

	10EM	10CEM-EM	10em-EM	SEMmax-EM	mean ML	std ML
10EM	0-0	3-86	39-39	23-64	-928.21	(13.69)
10CEM-EM	86-3	0-0	83-4	66-8	-919.85	(12.26)
10em-EM	39-39	4-83	0-0	29-56	-927.44	(13.96)
SEMmax-EM	64-23	8-66	56-29	0-0	-925.47	(13.05)

Table 9: Comparison of strategies by pairs with a large number of iterations for data set P3noise.

consists of two measurements: the duration (in minutes) of the eruption, and the waiting time (in minutes) before the next eruption. For those two examples we display, for the four mentioned selected strategies, the evolution of the maximum likelihood values with the number of iterations (Figures 2 and 4) and we depict both solutions in competition for each data set (Figures 3 and 5). The last example concerns 3641 observations in dimension five with no clear partitioning structure for which we consider a ten component Gaussian mixture. This data set concerns biological profiles of patients (see Sandor 1976). Figure 6 displays the maximum likelihood values for the four strategies as a function of the number of iterations.

As it appears from Monte Carlo experiments, a large number of iterations is required to ensure a sensible maximum likelihood value. Thus, we pay attention to experiments with $ITEMAX \geq 960$. Conclusions derived from noisy simulated data sets are confirmed: 10em-EM and 10CEM-EM perform the best in most cases. They provide similar results, but 10em-EM has a more stable behaviour. Otherwise, as for simulated data sets, there is no situation where the standard random strategy 10EM outperforms the other one strategies and, at best, strategy 10EM gives similar results than 10em-EM. Thus, the standard random strategy cannot be recommended.

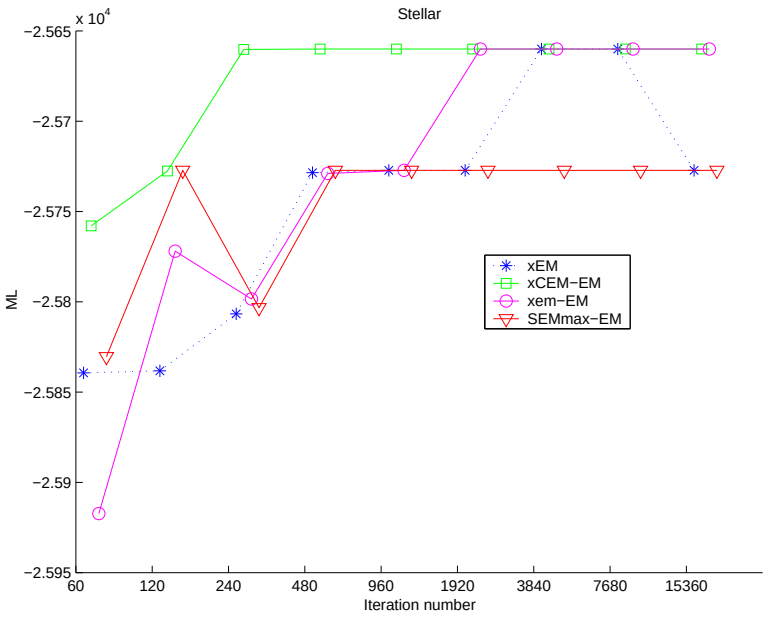
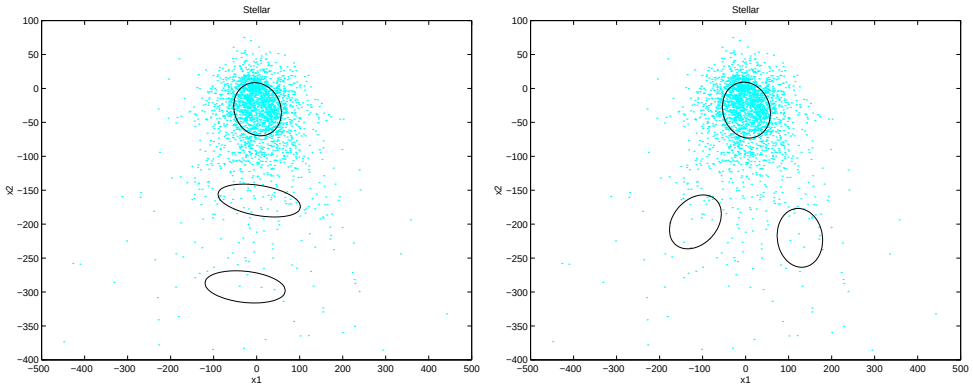


Figure 2: Evolution of the loglikelihood values with the number of iterations for the real data set "stars".



solution 1 solution 2

Figure 3: Two solutions in competition for the real data set "stars".

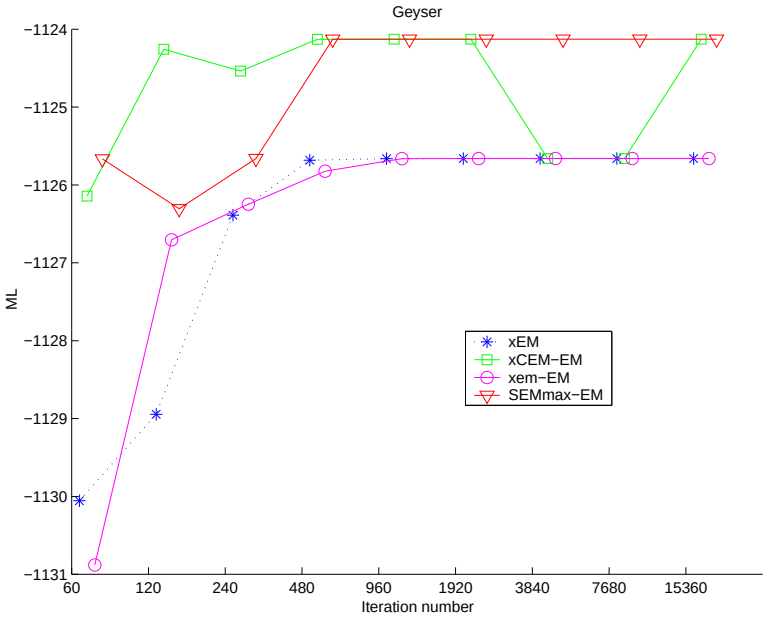
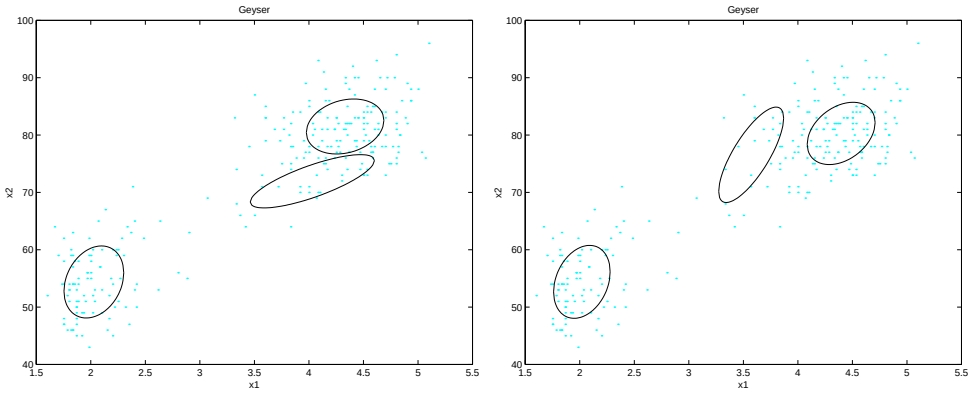


Figure 4: Evolution of the loglikelihood values with the number of iterations for the real data set "geyser".



solution 1 solution 2

Figure 5: Two solutions in competition for the real data set "geyser".

5 Conclusive remarks

We have presented and experimented simple strategies to deal with the important problem of getting a sensible maximum likelihood value when using the EM algorithm in mixture

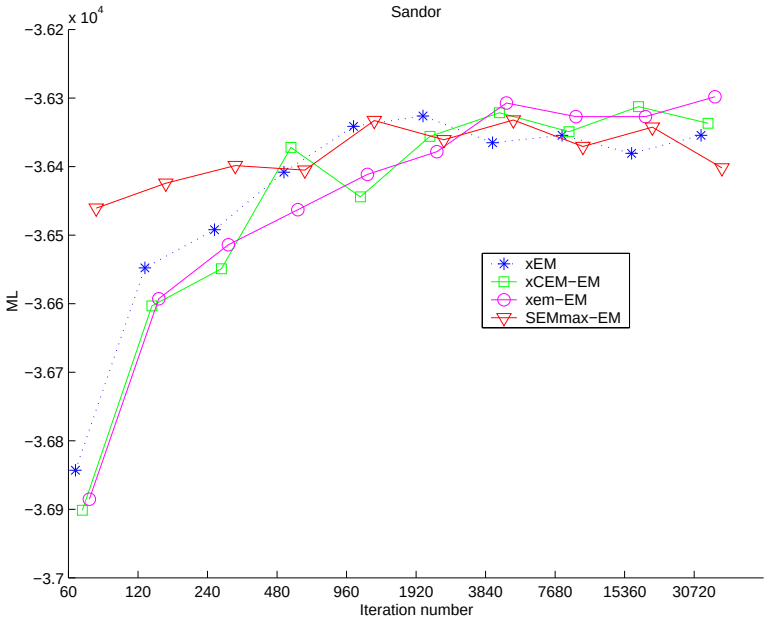


Figure 6: Evolution of the loglikelihood values with the number of iterations for the real data set "biological profiles".

inference. All those strategies are obtained by combining and repeating algorithms CEM, SEM and EM. From our experiments, the following comments can be made.

- For a good solution do not skimp on the number of iterations.
- The interest of repeated runs increases with the number of available iterations.
- There is no sensitive differences between the strategies.
- em-EM is maybe slighty better than the other ones and more stable.
- CEM-EM can perform well especially when a few iterations are available, but is the less stable strategy.
- Finally, we select em-EM as the default strategy in MIXMOD.

Otherwise, we can add the following remarks. In the strategies combining two algorithms we gave half iterations for both. From additional experiments not reported here, it appears that it is a good choice and there is little interest to choose other proportion for sharing the total number of iterations.

Here we focused on heuristics to get the highest likelihood value. But, in practice, especially when spurious local maximizers can occur, it may appear to be more interesting to select the local maximizer which has the largest attraction region because such a maximizer can be thought of as more stable. We do not dealt with this problem in the present paper, avoiding the possibility of spurious local maximizers in our experiments, but proposing heuristics to get stable maximizers deserves attention. From this point of view, it is possible that exploiting the stationary distribution of SEM can be of particular interest.

Appendix

In this appendix, it is proven that imposing equal volumes for the component variance matrices allow to avoid spurious local maximizers of the likelihood function.

The loglikelihood of an unrestricted variance matrices Gaussian mixture is unbounded as noted first by Day (1969). To see this, take $\mu_{k_0} = \mathbf{x}_{i_0}$ for some i_0 and k_0, fix all the other parameters except λ_{k_0} to some values and let λ_{k_0} tend to zero. In this case, it is easily seen that the loglikelihood converges to $+\infty$ as fast as $-\log\lambda_k$.

A simple way to avoid degeneracy is to impose equal variance matrix volumes. Actually, if the common volume λ tends to zero, this leads to a loglikelihood tending towards $-\infty$ as shown hereunder. Thus, degenerate solutions can not be reached when maximising the loglikelihood.

The mixture loglikelihood is

$$L(\theta|\mathbf{x}_1,\ldots,\mathbf{x}_n) = \sum_{i=1}^{n} \ln\left[\sum_{k=1}^{K} p_k \frac{1}{(2\pi)^{d/2}\lambda^{d/2}} \exp\Big(-\frac{\|\mathbf{x}_i-\mu_k\|^2}{2\lambda}\Big)\right], \tag{6}$$

with the norm $\|\mathbf{x}_i-\mu_k\|^2 = (\mathbf{x}_i-\mu_k)'(D_kA_kD_k')^{-1}(\mathbf{x}_i-\mu_k)$.

It can be written

$$L(\theta|\mathbf{x}_1,\ldots,\mathbf{x}_n) = \text{Cste} - n\frac{d}{2}\ln\lambda + \sum_{i=1}^{n} \ln\left[\sum_{k=1}^{K} p_k \exp\Big(-\frac{\|\mathbf{x}_i-\mu_k\|^2}{2\lambda}\Big)\right]. \tag{7}$$

From which it follows that

$$L(\theta|\mathbf{x}_1,\ldots,\mathbf{x}_n) < \text{Cste} - n\frac{d}{2}\ln\lambda - \frac{1}{2\lambda}R, \tag{8}$$

where

$$R = \sum_{i=1}^{n} \min_k \|\mathbf{x}_i-\mu_k\|^2.$$

Since the $\mathbf{x}_i$'s, arising for a Gaussian mixture, are all different with probability one, $R > 0$ with probability one. (In practical situations, $R > 0$ is ensured as soon as the sample $\mathbf{x}_1,\ldots,\mathbf{x}_n$ contains at leats $(k+1)$ different values.) Thus, the right hand side term of (8) tends to $-\infty$ as λ tends to zero and consequently $L(\theta|\mathbf{x}_1,\ldots,\mathbf{x}_n)$ tends to $-\infty$ as λ tends to zero.

References

Akaike, H. (1974), A new look at the statistical identification model. *IEEE Trans. Auto. Control*, **19**, 716-723.

Banfield, J. D. and Raftery, A. E. (1993), Model-based Gaussian and non Gaussian clustering. *Biometrics*, **49**, 803-821.

Celeux, G., Chauveau, D. and Diebolt, J. (1996), Stochastic Versions of the EM Algorithm: An Experimental Study in the Mixture Context. *Journal of Computational Statistical Computation and Simulation*, **55**, 287-314.

Celeux, G. and Govaert, G. (1992), A Classification EM Algorithm and two Stochastic Versions, *Computational Statistics and Data Analysis*, **14**, 315-332.

Celeux, G. and Govaert, G. (1993), Comparison of the Mixture and the Classification Maximum Likelihood in Cluster Analysis, *Journal of Computational Statistical Computation and Simulation*, **47**, 127-146.

Celeux, G. and Govaert, G. (1995), Gaussian Parsimonious Clustering Models, *Pattern Recognition*, **28**, 781-793.

Celeux, G., Chrétien, S., Forbes, F. and Mkhadri, A. (2001), A Component-wise EM Algorithm for Mixtures, *Journal of Computational and Graphical Statistics*, (to appear).

Day, N. E. (1969), Estimating the Components of a Mixture of Two Normal Distributions, *Biometrika*, **56**, 463-474.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977), Maximum Likelihood for Incomplete Data via the EM Algorithm (with discussion), *Journal of the Royal Statistical Society, B*, **39**, 1-38.

Liu, C. and Sun, D. X. (1997), Acceleration of EM Algorithm for Mixtures Models using ECME. *ASA Proceedings of The Stat. Comp. Session*, pp. 109-114.

McLachlan, G. J. and Basford, K. E. (1988), *Mixture Models: Inference and Applications to Clustering*. New York: Marcel Dekker.

McLachlan, G. J. and Krishnam, T. (1997), *The EM algorithm and Extensions*. New York: Wiley.

McLachlan, G. J. and Peel, D. (2000), *Finite Mixture Models*. New York: Wiley.

Meila, M. and Heckerman, D. (2001), An Experimental Comparison of Model-Based Clustering Methods, *Machine Learning*, **42**, 9-29.

Sandor, G. (1976), *Sémiologie biologique des protéines sériques*. Paris: Maloine.

Ueda, N. and Nakano, R. (1998), Deterministic Annealing EM Algorithm, *Neural Networks*, **11**, 271-282.

Wei, G. C. G. and Tanner, M. A. (1990), A Monte Carlo Implementation of the EM Algorithm and the Poor Man's Data Augmentation Algorithms, *Journal of the American Statistical Association*, **85**, 699-704.

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe - 38330 Montbonnot-St-Martin (France)

Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399